Class Balanced Adaptive Pseudo Labeling for Federated Semi-Supervised Learning

Ming Li Qingli Li Yan Wang*
Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University
lm1640362161@gmail.com, qlli@cs.ecnu.edu.cn, ywang@cee.ecnu.edu.cn

Abstract

This paper focuses on federated semi-supervised learning (FSSL), assuming that few clients have fully labeled data (labeled clients) and the training datasets in other clients are fully unlabeled (unlabeled clients). Existing methods attempt to deal with the challenges caused by not independent and identically distributed data (Non-IID) setting. Though methods such as sub-consensus models have been proposed, they usually adopt standard pseudo labeling or consistency regularization on unlabeled clients which can be easily influenced by imbalanced class distribution. Thus, problems in FSSL are still yet to be solved. To seek for a fundamental solution to this problem, we present Class Balanced Adaptive Pseudo Labeling (CBAFed), to study FSSL from the perspective of pseudo labeling. In CBAFed, the first key element is a fixed pseudo labeling strategy to handle the catastrophic forgetting problem, where we keep a fixed set by letting pass information of unlabeled data at the beginning of the unlabeled client training in each communication round. The second key element is that we design class balanced adaptive thresholds via considering the empirical distribution of all training data in local clients, to encourage a balanced training process. To make the model reach a better optimum, we further propose a residual weight connection in local supervised training and global model aggregation. Extensive experiments on five datasets demonstrate the superiority of CBAFed. Code will be available at https://github.com/mingllli/CBAFed.

1. Introduction

Federated learning (FL) aims to train machine learning models on a decentralized manner while preserving data privacy, i.e., separate local models are trained on separate local training datasets independently. In recent years, FL has received much attention for privacy protection reasons [32]. However, most of FL works focused on supervised learning with fully labeled data. But in practice, labeling process of large-scale training data is laborious and expensive. Due to the lack of funds or experts, large labeled training dataset is difficult for many companies and institutions to obtain. These may hinder applicability of FL.

To handle this problem, federated semi-supervised learning (FSSL) has been explored by many researchers recently [8, 14, 15]. There are broadly three lines of FSSL methods by considering different places and status of labeled data. The first two lines consider that there are only limited labeled data in the central server [8] or each client has partially labeled data [15]. The third line assumes that few clients have fully labeled data and the training datasets in other clients are fully unlabeled [14, 18, 31]. Our paper mainly focuses on the third line of FSSL, and we call local clients with fully labeled data as labeled clients and the other clients as unlabeled clients.

The main difficulties to train a third line FSSL model lie in three folds: 1) There are no labeled data in unlabeled clients. Thus, the training can be easily biased without label guidance. 2) Due to the divergent class distribution of labeled and unlabeled clients, namely the not independent and identically distributed data (Non-IID) setting, inaccurate supervisory signals may be generated in unlabeled clients via employing the model trained in labeled clients by either pseudo labeling or consistency regularization framework. 3) Due to the catastrophic forgetting problems in CNNs, with the training process of unlabeled clients going on, models may forget the knowledge learned on labeled clients and so decrease the prediction accuracy drastically.

RSCFed [14], the state-of-the-art method significantly boosts the FSSL performance by first distilling sub-consensus models, and then aggregating the sub-consensus models to the global model. The sub-consensus models can handle the Non-IID setting to some extent, but the mean-teacher based consistency regularization framework in unlabeled clients inevitably causes the accuracy degradation when the classes are imbalanced distributed. RSCFed at-

*Corresponding Author.

tempts to address the catastrophic forgetting problem by adjusting aggregation weights for labeled and unlabeled clients. But, it seems just alleviates the negative impact on the unlabeled clients and the problem is still yet to be tackled. Besides, the sub-consensus models may occur unstable training problems, and increase the communication cost. Other methods like FedConsist [31] and FedIRM [18] achieve promising results by utilizing pseudo labeling methods to produce artificial labels, but they do not consider the Non-IID setting among local clients.

Pseudo labeling methods are shown to be effective in semi-supervised learning (SSL) [13, 24, 33, 37], which generate pseudo-labels for unlabeled images and propose several strategies to ensure the quality of pseudo-labels. Existing FSSL works only adopt standard pseudo labeling or consistency regularization on FSSL. Since the mentioned difficulties hamper the usage of these methods, other remedy is usually proposed to alleviate the difficulties, which is palliative. Thus, we propose to study FSSL from the perspective of pseudo labeling, seeking for a fundamental solution to this problem.

Concretely, we present Class Balanced Adaptive Pseudo Labeling, namely CBAFed, by rethinking standard pseudo labeling methods in SSL. To handle the catastrophic forgetting problem, we propose a fixed pseudo labeling strategy, which builds a fixed set by letting pass informative unlabeled data and their pseudo labels at the beginning of the unlabeled client training. Due to the Non-IID and heterogeneous data partition problems in FL, training distribution of unlabeled data can be highly imbalanced, so existing thresholds are not suitable in FSSL. We design class balanced adaptive thresholds via considering the empirical distribution of all training data in local clients at the previous communication round. Analysis proves that our method sets a reasonably high threshold for extremely scarce classes and encourages a balanced training process. To enhance the learning ability and discover unlabeled data from tail classes, we propose to leverage information from so-called "not informative" unlabeled data. Besides, we also explore a novel training strategy for labeled clients and the central server, termed as residual weight connection, skip connecting weights from previous epochs (for labeled clients) or previous global models (for the central server). It can help the model reach better optimum, when the training distribution is imbalanced and training amount is small. We conduct extensive experiments on five datasets to show the effectiveness of CBAFed. Overall, our main contributions can be summarized as follows:

- We present a CBAFed method to deal with the catastrophic forgetting problems in federated learning . Unlike existing FSSL frameworks that directly adopts pseudo labeling or consistency regularization methods, CBAFed explores a fundamental solution to FSSL via designing a novel pseudo labeling method.
- We introduce a residual weight connection method, to improve the robustness of the models in labeled clients and the central server, which skip connects weights from previous epoch or communication round to finally reach better optimum.
- Experiments are conducted on five datasets: four natural datasets CIFAR-10/100, SVHN, fashion MNIST and one medical dataset ISIC 2018 Skin. CBAFed outperforms state-of-the-art FSSL methods by a large margin, with 11.3% improvements on SVHN dataset.

2. Related work

2.1. Federated Learning

Recently, federated learning is becoming more and more popular for privacy protection reasons. Statistical data heterogeneity is one of the most important research problems. FedAvg [19], one of the pioneer works to deal with this problem, performs weighted-averaging on local weights according to the local training size, has been used as the most widely recognized FL baseline. To deal with data heterogeneity, two research directions have been mostly explored: model aggregation [12, 27, 28] and local training [9, 11, 12, 17]. Recently, [23] demonstrates that self-attention-based architectures, e.g., vision transformers(ViT) are more robust to distribution shifts and can converge to better optimum over heterogeneous data.

2.2. Semi-supervised Leaning

The goal of standard semi-supervised learning (SSL) is to train a good model with limited labeled data and much more unlabeled data. Popular research directions involves consistency regularization [4, 26] and pseudo labeling [3, 10, 16]. Many recent works use the combinations of consistency regularization and pseudo labeling and achieves state-of-the-art performance [1, 2, 25, 30]. In this paper, we focus on using pseudo labeling method to deal with SSL. While these methods achieve much progress, they only consider pre-defined fixed threshold for pseudo labeling. FlexMatch [34] demonstrates that leveraging unlabeled data according to the model's learning status to flexibly adjust thresholds for different classes can boost performance. While these methods perform well in centralized SSL, they all update pseudo labels after every batch's update of the model, which is not suitable in FSSL as shown in later section.

2.3. Federated Semi-supervised Learning

Recently, with the development of federated learning, FSSL becomes popular. There are broadly three categories

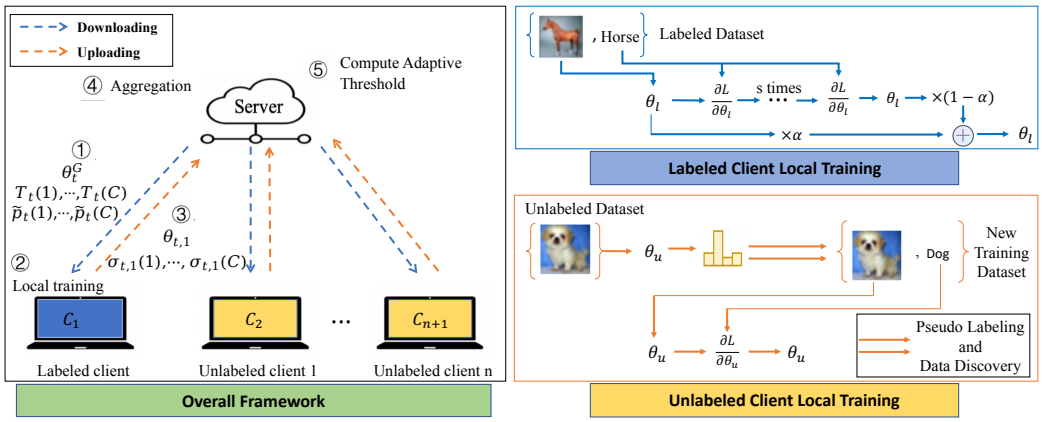

Figure 1. An overview of our CBAFed. In the central server (left side), the global model is aggregated with the returned local models (step ④) and the adaptive thresholds are calculated by the returned training data statistics (step ⑤). Then central server passes the global model, adaptive thresholds and class distribution to all local clients (step ①). After downloading these data, local clients perform local training on the right side (step ②). Labeled clients use labeled data to train the model with residual weight connection. Unlabeled clients obtain the new training dataset by adaptive pseudo labeling and tail class data discovery and use it to train the model. After local training, local clients return trained models and number of data in each class back to central server (step ③).

of FSSL. The first two categories assumes that every local client contains partially labeled data or there exists some labeled data in central server [8, 15]. The third category considers that few clients are fully labeled while other clients contain only unlabeled data. This paper focuses on the third category. To deal with this problem, Fed-Consist [31] and FedIRM [18] have been proposed. But, these two methods do not consider data heterogeneity in federated learning. Recently, RSCFed [14] proposes to perform random sub-sampling to reach consensus over clients. However, it uses standard consistency regularization for unlabeled data, which still suffers from the Non-IID setting. We emphasize that all the aforementioned approaches just use traditional pseudo labeling strategy, and none of them discuss how to apply pseudo labeling to FSSL.

3. Methodology

We first introduce the problem setting of federated semi-supervised learning. We consider decentralized training with labeled clients and unlabeled clients. Let m (usually $m = 1$) labeled clients be $\{C_1, C_2, \cdots, C_m\}$ and n (usually $n > 1$) unlabeled clients be $\{C_{m+1}, C_{m+2}, \cdots, C_{m+n}\}$. For labeled client C_ℓ, the labeled local training dataset is denoted as $D_\ell = \{(X_i^\ell, y_i^\ell)\}_{i=1}^{N_\ell}$, $\ell = 1, 2, \cdots, m$, where N_ℓ denotes the data number in D_ℓ. Similarly, for unlabeled client C_μ, the unlabeled local training dataset is denoted as $D_\mu = \{X_i^\mu\}_{i=1}^{N_\mu}, \mu = m+1, m+2, \cdots, n+m$.

The overall pipeline of the proposed CBAFed is shown in Fig. 1, which performs the following steps: (1) Warm up stage: train fully supervised models on only labeled clients using residual weight connection in a normal federated learning manner (Sec. 3.1) (2) The central server computes the empirical class distribution and obtains the class balanced adaptive thresholds, then passes them to local clients (Sec. 3.3); (3) All local clients update local models, adaptive threshold and class distribution. Labeled clients train local models on all the data using proposed residual weight connection; Unlabeled clients acquire the fixed training set (Sec. 3.2.2) by the threshold and the tail class datasets (Sec. 3.4), and train local models on the newly obtained training dataset. After local training, all local clients pass the trained model, the number of data in each class and the amount of training data to the central server; (4) The central server aggregates a new model with residual weight connection, computes the class distribution, and obtains the class balanced adaptive threshold. Then, the central server passes them to local clients; (5) Repeat step (3)-(4) until the specified number of communication round is reached. The overall training procedure of our CBAFed is provided in Algorithm 1 in the supplementary material.

3.1. Residual Weight Connection

We propose a novel training method in local supervised training and global model aggregation called residual weight connection. Our idea of res-weight is simple and similar to ResNet. In ResNet, there is a skip connection between every layer. For our res-weight connection, there is a skip connection of model's parameters between training epochs (or communication rounds). Let θ^E be the parameters of model after the training epoch (communication round) $E(E > 0)$, then the final parameters of model after this training epoch by using residual weight connection will

be

$$\theta^E = \begin{cases} \theta^E & E\%s \neq 0 \\ \alpha_1 \theta^{E-s} + (1-\alpha_1)\theta^E & E\%s = 0 \end{cases} \quad (1)$$

where α_1 is the parameter to scale the size of θ^{E-s} and θ^E and s is the number of skip epoch. In [22], authors show that averaging model weights over training steps tends to produce a more accurate model than using the final weights directly. In our experiments, we will show that this training method can effectively make the model more robust and finally reach better optimum, enhancing the predicted accuracy of pseudo labeling. Note that although the formula of our residual weight connection is similar to mean teachers [26], the update strategy and usage scenario is very different from it. Please refer to supplementary material for more detailed comparison and discussions.

3.2. Pseudo Labeling Methods

3.2.1 Review of Classic Batch-based Pseudo Labeling

In semi-supervised learning, for unlabeled data, traditional approaches [2, 25, 34], such as Fixmatch [25] and Flexmatch [34], use original data or their weak augmented version in one **batch** to generate pseudo labels. These labels are adopted to supervise model's training. For detailed review, please refer to supplementary materials.

3.2.2 Fixed Pseudo Labeling in FSSL

Warm up stage. To conduct pseudo labeling for unlabeled data, in the warm up stage, we train our model only with labeled data. In the first P communication round, training only conducted in labeled clients. In local training on labeled client C_ℓ, the training loss is defined as:

$$\mathcal{L}_\ell = \frac{1}{|D_\ell|} \sum_{(X^\ell, y^\ell) \in D_\ell} H(y^\ell, p_m(y|\theta_t^\ell(X^\ell))), \quad (2)$$

where θ_t^ℓ means local model in labeled client C_ℓ at communication round t. The local training epoch is set to $J(J > 1)$ with residual weight conenction. Then we use FedAVG [19] with residual weight connection for model aggregation:

$$\theta_{t+1}^G = \begin{cases} \sum_{\ell=1}^m \frac{|D_\ell|}{\sum_{i=1}^m |D_i|} \theta_t^\ell & t\%s \neq 0 \\ \alpha_1 \theta_{t+1-s}^G + (1-\alpha_1) \sum_{\ell=1}^m \frac{|D_\ell|}{\sum_{i=1}^m |D_i|} \theta_t^\ell & t\%s = 0 \end{cases} \quad (3)$$

where θ_{t+1}^G is the global model of communication round $t+1$. Note that, usually, the number of labeled clients is only 1, so we can finish warm up stage in only one communication round(training can be completed on one labeled client).

Different from traditional semi-supervised learning, in federated semi-supervised learning, Non-IID data partitions in clients can easily lead to catastrophic forgetting [7, 21, 23]. Thus, local models trained on unlabeled clients usually forget the former learned knowledge by batch-based pseudo labeling method, which will lead to abrupt degradation and sabotage the training of unlabeled clients. To handle heterogeneous data partitions and catastrophic forgetting, we propose a new pseudo labeling method called fixed pseudo labeling. For the unlabeled training dataset D_μ in an unlabeled client C_μ, after initializing the local model θ_t^μ with the global model θ_t^G, we first compute the pseudo label for each data X_i^μ in D_μ via:

$$\hat{y}_i^\mu = \arg\max p_m(y|\theta_t^\mu(X_i^\mu), i = 1, 2, \cdots, N_\mu. \quad (4)$$

We obtain a subset of unlabeled data with their pseudo labels as supervisory signals during local client training:

$$\widetilde{\mathcal{D}}_\mu = \{(X_i^\mu, \hat{y}_i^\mu) \mid X_i^\mu \in D_\mu \wedge \max(p_m(y|\theta_t^\mu(X_i^\mu))) > \tau\}_{i=1}^{N_\mu}. \quad (5)$$

Then we use this fixed set $\widetilde{\mathcal{D}}_\mu$ as the training dataset and define the loss to train our unlabeled client as: $\mathcal{L}_\mu = \frac{1}{|\widetilde{\mathcal{D}}_\mu|} \sum_{(X^\mu, \hat{y}^\mu) \in \widetilde{\mathcal{D}}_\mu} H(\hat{y}^\mu, p_m(y|\theta_t^\mu(X^\mu)))$.

3.3. Class Balanced Adaptive Threshold for Pseudo Labeling

In semi-supervised learning, the value of threshold τ is non-trivial for selecting confident pseudo labels, namely informative unlabeled data. Traditionally, a fixed threshold is pre-defined [25], but setting a fixed threshold usually makes the model fail to consider different learning status and learning difficulties of different classes. In one of the most recent works, Curriculum Pseudo Labeling [34] is proposed to achieve state-of-the-art results on most semi-supervised learning benchmarks, where a flexible threshold for class c at time step t is calculated by: $\mathcal{T}_t(c) = \beta_t(c) \cdot \tau$, where $\beta_t(c)$ is the ratio of the number of selected pseudo labels in class c and the maximum number of selected pseudo labels of all classes. $\beta_t(c)$ is used to scale the fixed threshold τ. Curriculum Pseudo Labeling significantly boosts the accuracy and convergence performance of various semi-supervised learning algorithms. But in FSSL, there exist heterogeneous data partition problems. Due to the Non-IID partition, the labeled data are not balanced, so purely using the number of selected unlabeled data to design threshold is improper. Moreover, the data of some classes can be extremely rare in certain clients, which will lead to a very low $\beta_t(c)$, and thus, a very low threshold ($\mathcal{T}_t(c) \approx 0$). In other words, directly applying Curriculum Pseudo Labeling will introduce many noisy labels into training.

To this end, we propose a novel Class Balanced Adaptive threshold for Pseudo Labeling (CBAPL). For an unlabeled client C_μ in the tth communication round, let the number of selected pseudo labels in class c be:

$$\sigma_t^\mu(c) = \sum_{i=1}^{N_\mu} \mathbf{1}(\max(p_m(y|\theta_t^\mu(X_i^\mu))) > \mathcal{T}_t(c)) \mathbf{1}(\hat{y}_i^\mu = c), \quad (6)$$

where $\mathbf{1}(\cdot)$ is the indicator function. The data number of class c in labeled client C_ℓ is calculated by: $\sigma_t^\ell(c) = \sum_{i=1}^{N_\ell} \mathbf{1}(y_i^\ell = c)$. After local training, $\sigma_t^\mu(c)$, $\sigma_t^\ell(c)$ and weights of local models are returned to central server. Then the central server calculates the total number of training data of class c at communication round t:

$$\sigma_t(c) = \sum_{\ell=1}^{m} \sigma_t^\ell(c) + \sum_{\mu=m+1}^{n+m} \sigma_t^\mu(c). \tag{7}$$

We then normalize the training numbers and acquire the empirical distribution of training data:

$$\widetilde{p}_t(c) = \frac{\sigma_t(c)}{\sum_{i=1}^{C} \sigma_t(i)}. \tag{8}$$

The standard deviation of the empirical distribution is calculated by:

$$std(\widetilde{p}_t) = \sqrt{\frac{1}{C-1}\sum_{c=1}^{C}(\widetilde{p}_t(c) - \overline{p}_t)^2}, \tag{9}$$

where $\overline{p}_t = \frac{1}{C}\sum_{c=1}^{C} \widetilde{p}_t(c)$. Lastly, the threshold of class c is calculated by

$$\tau_{t,c} = \widetilde{p}_t(c) + \tau - std(\widetilde{p}_t), \tag{10}$$

where τ is the pre-defined threshold base. We then set an upper bound of threshold as:

$$\mathcal{T}_{t+1}(c) = \begin{cases} \tau_{t,c}, & \tau_{t,c} < \tau_h \\ \tau_h, & \tau_{t,c} \geq \tau_h \end{cases}, \tag{11}$$

where τ_h is the pre-defined upper bound of threshold (usually $\tau_h = 0.95$ for ensuring a certain amount of training data in unlabeled clients). If our model is trained with more data in one class, the threshold of that class will be higher since $\widetilde{p}_t(c)$ is higher. $std(\widetilde{p}_t)$ is important for balancing the empirical distribution of training data. Please refer to the supplementary material for more discussions.

We argue that for scarce classes, unlike Curriculum Pseudo Labeling [34] whose threshold is extremely low, our CBAPL calculates a threshold with a relatively high lower bound. We have theorem below:

Theorem 3.1.

$$\tau + \widetilde{p}_t(c) - \sqrt{\frac{1}{C}} \leq \mathcal{T}_t(c) \leq \tau + \widetilde{p}_t(c), \tag{12}$$

Proof is provided in the supplementary material.

Since $\tau >> \sqrt{\frac{1}{C}}$, $\mathcal{T}_t(c)$ will have a high lower bound ($>>$ 0). To balance the trade-off between $\widetilde{p}_t(t)$ and $\tau - \sqrt{1/C}$, a normalization factor is applied to consider the class number. Thus, we propose a modified $\widetilde{p}_t(c)$ as below:

$$\widetilde{p}_t(c) = \frac{\sigma_t(c)}{\sum_{i=1}^{C} \sigma_t(i)} \times \frac{C}{10}. \tag{13}$$

At the first P communication rounds, the models are trained on only labeled clients. Thus, when $t \in \{1, ..., P\}$, $\sigma_t^\mu(c) = 0$ for all classes. After calculating the thresholds for all classes via Eq. 11, the central server will pass the thresholds to local clients. Unlabeled clients further obtain the fixed pseudo label training dataset by:

$$\begin{aligned} \widetilde{\mathcal{D}}_{t+1,\mu} = \{(X_i^\mu, \hat{y}_i^\mu) | X_i^\mu \in D_\mu \\ \wedge \max(p_m(y|\theta_t^\mu(X_i^\mu))) > \mathcal{T}_{t+1}(\hat{y}_i^\mu)\}_{i=1}^{N_\mu}. \end{aligned} \tag{14}$$

3.4. Discovery of Unlabeled Data from Tail Classes

In FSSL, heterogeneous data partitions will lead to class imbalanced distribution in local clients and there are rare or even no data from some classes in certain clients. For warm up stage in labeled clients, it is similar to long-tailed classification, so the problems in long-tailed classification will also exist in our pseudo labeling process, i.e., models tend to classify tail (rare) classes as head (common) classes [29, 35, 36]. Although adaptive pseudo labeling can lower the threshold of tail classes, much less data will be selected by PL since very few data from tailed classes will be classified as correct classes. To enhance the learning ability and discover unlabeled data from these classes, instead of directly dropping the data whose maximum confidences are low, namely, "not informative" unlabeled data, we propose to leverage information from them. For simplicity, we drop subscript t and default all representations occur in tth communication round in Sec. 3.4. Firstly, we define a mask function $\mathcal{M}(\cdot)$: $\mathcal{R}^C \to \mathcal{R}^C$ for a probability simplex $p \in \mathcal{R}^C$:

$$\mathcal{M}_i(p) = \begin{cases} p_i & i \neq \arg\max p \\ 0 & i = \arg\max p \end{cases}, \tag{15}$$

where p_i is the ith element in p. The effect of mask function $\mathcal{M}(\cdot)$ is to set the maximum of a probability simplex to 0. Then, we turn to analyze the second largest confidence score via:

$$\begin{aligned} D_\mu^{tail} = \{(X_i^\mu, \hat{y}_i^{\mu\prime}) | X_i^\mu \in D_\mu \\ \wedge \max(p_m(y|\theta_t^\mu(X_i^\mu))) \leq \mathcal{T}_t(\hat{y}_i^\mu) \wedge \widetilde{p}_t(\hat{y}_i^{\mu\prime}) < \frac{\beta}{C}\}, \end{aligned} \tag{16}$$

where $\hat{y}_i^{u\prime} = \arg\max \mathcal{M}(p_m(y|\theta_t^\mu(X_i^\mu)))$ and β is a hyperparameter. In Eq. 15, the parameter β is to find the tail classes ("not informative" unlabeled data). For unlabeled data X_i^μ, if its largest confidence is smaller than threshold

(not informative or confident, *i.e.*, $\max(p_m(y|\theta_t^\mu(X_i^\mu))) \leq \mathcal{T}_t(\hat{y}_i^\mu)$) and and the class of second largest confidence is tail class (*i.e.*, $\widetilde{p}_t(\hat{y}_i^{\mu\prime}) < \frac{\beta}{C}$), we regard it as misclassfied data and include it in training with the class of second largest confidence as its label. Finally, the new training dataset for unlabeled client C_μ is defined as

$$D_\mu^{train} = D_\mu^{tail} \cup \widetilde{D}_\mu, \tag{17}$$

and the final training loss in C_μ is rewritten as

$$\mathcal{L}_\mu = \frac{1}{|D_\mu^{train}|} \sum_{(X^\mu, \hat{y}^\mu) \in D_\mu^{train}} H(\hat{y}^\mu, p_m(y|\theta_t^\mu(X^\mu))). \tag{18}$$

The calculation of $\sigma_t^u(c)$ is revised as:

$$\sigma_t^\mu(c) = \sum_{(X^\mu, \hat{y}^\mu) \in D_\mu^{train}} \mathbf{1}(\hat{y}^\mu = c). \tag{19}$$

3.5. Aggregation of local models

Since not all data in local clients are included into training, we propose to use only data included in training for model aggregation. The scale factor w_t^i is computed by

$$w_t^i = \begin{cases} \frac{|D_i|}{|D_t^{train}|} & \text{if } i \in \{1, \cdots, m\} \\ \frac{|D_{t,i}^{train}|}{|D_t^{train}|} & \text{if } i \in \{m+1, \cdots, m+n\}, \end{cases} \tag{20}$$

where $|D_t^{train}| = \sum_{\ell=1}^m |D_\ell| + \sum_{\mu=m+1}^{m+n} |D_{t,\mu}^{train}|$. Then at $(t+1)$th synchronization round, the global model θ_{t+1}^G is computed by

$$\theta_{t+1}^G = \begin{cases} \sum_{i=1}^{m+n} w_t^i \theta_t^i & t\%s \neq 0 \\ \alpha_2 \theta_{t+1-s}^G + (1-\alpha_2) \sum_{i=1}^{m+n} w_t^i \theta_t^i & t\%s = 0, \end{cases} \tag{21}$$

where α_2 is a hyperparameter.

4. Experiments

4.1. Experimental Setup

Benchmark Datasets To evaluate the effectiveness of our proposed method, we conduct extensive experiments on four image classification datasets, i.e., SVHN, CIFAR-10, CIFAR-100, Fashion MNIST and one medical image classification dataset: ISIC 2018 (Skin Lesion Analysis Towards Melanoma Detection). Dataset splitting and pre-processing are provided in the supplementary material.

FSSL Setting Simulation Following [14], training datasets contain 10 clients: one labeled and nine unlabeled. We use a Dirichlet distribution $Dir(\gamma)$, where $\gamma = 0.8$ for five datasets [14] to generate Non-IID data partition in clients.

Implementation Details We utilize the SGD optimizer with a momentum of 0.9, and implement our method with PyTorch. We adopt ResNet18 [6] from PyTorch for all datasets. For fair comparison, we use the same network architecture and training protocol, including the optimizer, data preprocessing, etc. across all FSSL methods. The local training epoch is set to 11 for labeled clients and 1 for unlabeled clients. More implementation details are provided in supplementary material.

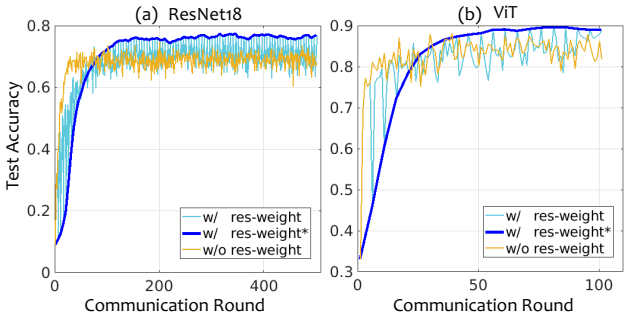

Figure 2. Test accuracy curves in local training of SVHN dataset w/ and w/o residual weight connection. ResNet18 (a) and ViT (b) are adopted as the backbones. W/ res-weight* indicates we only show test accuracy on epochs (communication rounds, since local training epoch for labeled client is 1) with skip weight connection. Best viewed electronically.

Comparisons We compare our method against state-of-the-art FSSL methods, including RSCFed [14], FedIRM [18], and Fed-Consist [31]. Since FedIRM [18] and Fed-Consist [31] are not desgined for dealing with Non-IID FSSL, following [14], we enlarge the weight of labeled client to about 50% and other nine unlabeled clients share the remaining 50% weight in each FSSL synchronization round, when implementing these methods. We also compare our network against FedAvg [19] trained with all clients as an upper bound and trained with only one labeled clients as the lower bound. Besides, to show the effectiveness of our residual weight connection, we report the result of FedAvg [19] trained with only one labeled clients using residual weight connection. Since Fed-Consist [31] utilizes traditional batch-based pseudo labeling method, we report the result of Fed-Consist [31] using our proposed fixed pseudo labeling without enlarging the weight of labeled client.

4.2. Comparisons with State-of-the-arts

Comparison Results. Table 1 reports the quantitative results of our CBAFed and other state-of-the-art methods on five benchmarks. We can observe that our proposed CBAFed achieves superior performances over all competitors on the five benchmark datasets. For some easy tasks (e.g., SVHN and Fashion-MNIST), the performance of our CBAFed is approaching upper bound obtained by FedAVG [19] and surpasses all state-of-the-arts by a large margin, showing a strong power of fixed pseudo labeling and class balanced threshold. Note that if replacing the traditional

Table 1. Results on SVHN, CIFAR-10/100, Fashion MNIST and ISIC 2018 datasets under heterogeneous data partition with ResNet18. FedAVG$^+$ means FedAvg [19] trained with all one labeled clients using our residual weight connection. Fed-consist$^+$ means Fed-Consist [31] using our proposed fixed pseudo labeling without enlarging the weight of labeled client.

Labeling Strategy	Method	Client Num.		Dataset				
		labeled	unlabeled	SVHN	CIFAR10	CIFAR100	Fashion-MNIST	ISIC 2018
Fully supervised	FedAvg [19](upper-bound)	10	0	91.83	80.89	51.38	90.14	81.32
	FedAvg [19](lower-bound)	1	0	67.71	54.66	20.49	74.87	65.13
	FedAvg$^+$ [19]	1	0	76.98	58.21	24.84	78.26	66.69
Semi supervised	FedIRM [18]	1	9	69.22	52.84	20.20	76.83	64.85
	Fed-Consist [31]	1	9	70.56	54.23	21.81	76.57	65.20
	Fed-Consist$^+$ [31]	1	9	86.57	56.35	23.25	78.35	65.50
	RSCFed [14]	1	9	76.74	57.07	28.46	78.40	67.21
	CBAFed(ours)	1	9	**88.07**	**67.08**	**30.18**	**85.49**	**68.29**

batch-based pseudo labeling by our fixed pseudo labeling in Fed-Consist [31], the performance is largely increased. It shows that fixed pseudo labeling method is well suited for Non-IID FSSL.

Trained on one labeled client, FedAVG [19] w/ residual weight connection achieves much better performance compared with that w/o residual weight connection. Fig. 2 (a) shows the test accuracy curves during training. Due to the imbalanced training data distribution in labeled client, the test accuracy is unstable during training for FedAVG (w/o res-weight in the figure). But, if training w/ our residual weight connection, the test accuracy curve is much more stable and the performance is also enhanced (w/ res-weight and w/ res-weight*). To sum up, residual weight connection can improve the robustness of the models when the training amount is small and the training distribution is imbalanced. Since long tailed vision recognition task is similar to local training in labeled client, it is a potential direction to use our residual weight connection in long tailed vision. Bisides, we also conduct experiments on partially labeled clients, results are shown in the supplementary material.

ViT Backbone. Vision Transformer(ViT) has shown to be more robust to heterogeneous data and distribution shifts, and has been demonstrated to converge faster with better optimum in Federated learning [23]. To further study the effect of our method, we conduct experiments on SVHN dataset using ViT-Tiny as backbone for all competitors. The implementation details are provided in the supplementary material. Table 2 shows the comparison results. Our method can outperform all other methods. Similarly, trained on one labeled client, FedAVG [19] w/ residual weight connection surpasses the one w/o residual weight connection, meaning that our residual weight connection is also effective on ViT, not only CNNs, as shown in Fig. 2 (b).

Two Labeled Clients. To better understand the effectiveness of our CBAFed and study how to use residual weight connection when the number of labeled clients is more than one, following [14, 18], we conduct experiments on CIFAR-10 by dividing the whole training data into 10 clients, with two labeled clients and eight unlabeled clients. Results are reported in Table 3. Our method has a better performance compared with all state-of-the-art FSSL methods. Adding our residual weight connection and fixed pseudo labeling lead to much performance gain. In the supplementary material, we also discuss different training strategies used for labeled clients w.r.t. number of local epochs and the usage of residual weight connection.

Table 2. Comparison of our method against RSCFed [14], Fed-Consist [31] and FedAVG [19] in SVHN dataset on ViT [5] as the backbone, with one labeled and nine unlabeled clients.

Method	Client Num.		Accuracy
	labeled	unlabeled	
FedAVG [19](upper bound)	10	0	96.81
FedAVG [19](lower bound)	1	0	81.68
FedAVG$^+$ [19]	1	0	88.93
FedIRM [18]	1	9	79.44
Fed-Consist [31]	1	9	85.91
Fed-Consist$^+$ [31]	1	9	93.21
RSCFed [14]	1	9	89.43
CBAFed(ours)	1	9	**95.09**

Table 3. Comparison of our method against RSCFed [14], Fed-Consist [31], FedIRM [18] and FedAVG [19] with the number of labeled and unlabeled client set to 2 and 8.

Method	Client Num.		Accuracy
	labeled	unlabeled	
FedAVG [19](upper bound)	10	0	80.89
FedAVG [19](lower bound)	2	0	61.85
FedAVG$^+$ [19]	2	0	66.55
FedIRM [18]	2	8	62.62
Fed-Consist [31]	2	8	61.67
Fed-Consist$^+$ [31]	2	8	68.04
RSCFed [14]	2	8	64.25
CBAFed(ours)	2	8	**72.01**

Communication Cost. Since the statistics of local training data can be nearly neglected (very low compared with model's parameter), our method does not add any burden in communication cost, which is only 66% of the state-of-the-art method RSCFed [14].

4.3. Study of Pseudo Labeling Strategies in FSSL

In this section, we discuss how to perform pseudo labeling in FSSL and why traditional batch-based pseudo la-

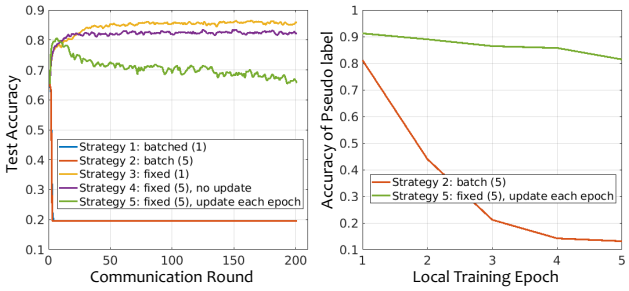

Figure 3. Left: Test accuracy of all 5 strategies after every communication round. Note that the test accuracy of communication round 0 is the test accuracy of model trained on labeled client. Right: Accuracy of pseudo labels in local training epoch of one randomly selected unlabeled client. Best viewed electronically.

beling fails. We conduct a simple experiment on SVHN dataset [20] using ResNet18 [6]. The experimental setting is the same as illustrated in Sec. 4.1 except that we only use pseudo labeling without other proposed modules. We consider five training strategies: (1) Batched pseudo labeling (1 local training epoch). (2) Batched pseudo labeling (5 local training epochs). (3) Fixed pseudo labeling (1 local training epoch). (4) Fixed pseudo labeling (5 local training epochs), and we do not update the fixed training dataset in all local training epochs after pseudo labeling. (5) Fixed pseudo labeling (5 local training epochs), and we update the fixed training dataset at the beginning of every training epoch. For labeled client, the local training epoch is set to 1 for all 5 strategies. The total communication round is set to 200 for all strategies. The test accuracy after communication round is shown in the left figure of Fig. 3. We can observe the following: **i)**, Batch-based pseudo labeling performs poorly in FSSL. It is known that CNNs often do not work well on out-of-distribution data [23]. Thus, a few update can result in catastrophic forgetting on Non-IID data partitions. The right figure of Fig. 3 shows the predicted accuracy of communication round one in five local training epochs of strategy (2) and (5). **ii)**, Our fixed pseudo labeling method is capable of dealing with catastrophic forgetting. Strategy (3) performs the best and strategy (5) performs much worse than strategy (3) and (4), indicating that one local training epoch is optimum. **iii)** Comparing strategy (3) with (4), the former has better performance but the latter has faster convergence speed. It means more local training epochs will speed up convergence but may degrade performance, similar phenomenon is also found in [23].

4.4. Ablation Study

We conduct ablation study to demonstrate the effectiveness of the main components of our method: residual weight connection, class balanced pseudo labeling, and tail class data discovery. Table 4 shows the results on three datasets: CIFAR-10/100 and Fashion-MNIST. It can be seen that all three modules are important for boosting performance. Then, we aim at discussing hyper-parameters of threshold base τ, upper bound threshold τ_h, and parameters for selecting tail class data β. Experiments are conducted on Fashion-MNIST. As shown in Fig. 4, performances are not sensitive within certain ranges.

Table 4. Ablation Study of CBAFed in CIFAR-10/100 and Fashion MNIST Datasets. Fixed PL: fixed pseudo labeling, CBA: class balanced adaptive pseudo labeling, DD: tail class data discovery.

Dataset	Fixed PL	CBA	DD	Res-Weight	Accuracy
CIFAR-10	✓				59.16
	✓	✓			64.29
	✓	✓	✓		65.15
	✓	✓	✓	✓	67.08
CIFAR-100	✓				27.64
	✓	✓			29.41
	✓	✓	✓		29.86
	✓	✓	✓	✓	30.18
Fashion-MNIST	✓				79.99
	✓	✓			80.87
	✓	✓	✓		84.37
	✓	✓	✓	✓	85.49

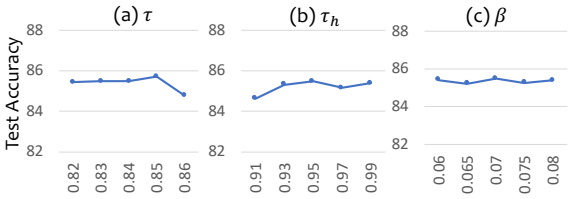

Figure 4. Performance changes on Fashion-MNIST by varying (a) threshold base τ and (b) upper bound threshold τ_h, and (c) parameter for selecting tail class data β.

5. Conclusion

We present Class Balanced Adaptive pseudo labeling (CBAFed) for Federated Semi-Supervised Learning (FSSL). In CBAFed, a fixed pseudo labeling strategy is proposed to handle the catastrophic forgetting problem. To deal with the Non-IID setting of FSSL, we propose a class balanced adaptive thresholds selection method to choose better pseudo labels. Furthermore, a residual weight connection method is designed to make the model reach better optimum. We evaluate CBAFed on five datasets, whose performances show the superiority of our method.

Limitations. Like other FSSL methods, a good model in warm-up cannot be guaranteed if the number of data in labeled clients are extremely few, so the final global model may not perform well.

6. Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant No. 62101191), Shanghai Natural Science Foundation (Grant No. 21ZR1420800), and the Science and Technology Commission of Shanghai Municipality (Grant No. 22DZ2229004).

References

[1] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019. 2

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019. 2, 4

[3] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2613–2622, 2021. 2

[4] Wenhui Cui, Yanlin Liu, Yuxing Li, Menghao Guo, Yiming Li, Xiuli Li, Tianle Wang, Xiangzhu Zeng, and Chuyang Ye. Semi-supervised brain lesion segmentation with an adapted mean teacher model. In *International Conference on Information Processing in Medical Imaging*, pages 554–565. Springer, 2019. 2

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 7

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6, 8

[7] Wenke Huang, Mang Ye, and Bo Du. Learn from others and be yourself in heterogeneous federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10143–10153, 2022. 4

[8] Wonyong Jeong, Jaehong Yoon, Eunho Yang, and Sung Ju Hwang. Federated semi-supervised learning with inter-client consistency & disjoint learning. In *Proc. ICLR*, 2021. 1, 3

[9] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020. 2

[10] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 2

[11] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021. 2

[12] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020. 2

[13] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, and Pheng-Ann Heng. Transformation consistent self-ensembling model for semi-supervised medical image segmentation. *IEEE Trans. Neural Networks and Learning Systems*, 2020. 2

[14] Xiaoxiao Liang, Yiqun Lin, Huazhu Fu, Lei Zhu, and Xiaomeng Li. Rscfed: Random sampling consensus federated semi-supervised learning. In *Proc. CVPR*, 2022. 1, 3, 6, 7

[15] Haowen Lin, Jian Lou, Li Xiong, and Cyrus Shahabi. Semifed: Semi-supervised federated learning with consistency and pseudo-labeling. *arXiv preprint arXiv:2108.09412*, 2021. 1, 3

[16] Fengbei Liu, Yu Tian, Yuanhong Chen, Yuyuan Liu, Vasileios Belagiannis, and Gustavo Carneiro. Acpl: Anti-curriculum pseudo-labelling for semi-supervised medical image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20706, 2022. 2

[17] Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. Feddg: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1013–1023, 2021. 2

[18] Quande Liu, Hongzheng Yang, Qi Dou, and Pheng-Ann Heng. Federated semi-supervised medical image classification via inter-client relation matching. In *Proc. MICCAI*, 2021. 1, 2, 3, 6, 7

[19] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 2, 4, 6, 7

[20] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011. 8

[21] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022. 4

[22] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992. 4

[23] Liangqiong Qu, Yuyin Zhou, Paul Pu Liang, Yingda Xia, Feifei Wang, Ehsan Adeli, Li Fei-Fei, and Daniel Rubin. Rethinking architecture design for tackling data heterogeneity in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10061–10071, 2022. 2, 4, 7, 8

[24] Yinghuan Shi, Jian Zhang, Tong Ling, Jiwen Lu, Yefeng Zheng, Qian Yu, Lei Qi, and Yang Gao. Inconsistency-aware uncertainty estimation for semi-supervised medical image segmentation. *IEEE Trans. Medical Imaging*, 41(3):608–620, 2022. 2

[25] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence.

Advances in neural information processing systems, 33:596–608, 2020. 2, 4

[26] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 2, 4

[27] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*, 2020. 2

[28] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020. 2

[29] Yidong Wang, Bowen Zhang, Wenxin Hou, Zhen Wu, Jindong Wang, and Takahiro Shinozaki. Margin calibration for long-tailed visual recognition. *arXiv preprint arXiv:2112.07225*, 2021. 5

[30] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268, 2020. 2

[31] Dong Yang, Ziyue Xu, Wenqi Li, Andriy Myronenko, Holger R Roth, Stephanie Harmon, Sheng Xu, Baris Turkbey, Evrim Turkbey, Xiaosong Wang, et al. Federated semi-supervised learning for covid region segmentation in chest ct using multi-national data from china, italy, japan. *Medical image analysis*, 70:101992, 2021. 1, 2, 3, 6, 7

[32] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019. 1

[33] Lequan Yu, Shujun Wang, Xiaomeng Li, Chi-Wing Fu, and Pheng-Ann Heng. Uncertainty-aware self-ensembling model for semi-supervised 3d left atrium segmentation. In *Proc. MICCAI*, 2019. 2

[34] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021. 2, 4, 5

[35] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2361–2370, 2021. 5

[36] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9719–9728, 2020. 5

[37] Yuyin Zhou, Yan Wang, Peng Tang, Song Bai, Wei Shen, Elliot K. Fishman, and Alan L. Yuille. Semi-supervised 3d abdominal multi-organ segmentation via deep multi-planar co-training. In *Proc. WACV*, 2019. 2